The HEP-MATH package*

Extended math macros

Jan Hajer†

2023/07/01

Abstract

The HEP-MATH package provides some additional features beyond the MATHTOOLS and AMSMATH packages.

To use the package place `\usepackage{hep-math}` in the preamble.

The MATHTOOLS [1] package is loaded, which in turn loads the $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX AMSMATH [2] package. Horizontal spacing in inline equations and page breaks in block equations are marginally adjusted.

`\left` `\right` Spacing around `\left` and `\right` is fixed with the MLEFTRIGHT package [3].

1 Macros

`\mathdef` The `\mathdef{`⟨*name*⟩`}[`⟨*arguments*⟩`]{`⟨*code*⟩`}` macro (re-)defines macros only within math mode without changing the text mode definition.

`\i` The imaginary unit `\i` and the differential `\d` are defined using this functionality.

`\d` `\overline` The `\overline` macro is adjusted to $\overline{\text{work also outside}}$ of math mode using the SOULUTF8 [4] package.

`\oset` A better looking over left right arrow is defined i.e. $\overleftrightarrow{\partial}$ using a new `\oset{`⟨*over*⟩`}{`⟨*math*⟩`}` functionality.

`\overleft` `\overright` `\overleftright` Diagonal matrix `\diag`, signum `\sgn`, trace `\tr`, `\Tr`, and `\rank` operators are defined.

The real and imaginary projectors are redefined to look like ordinary operators.

`\diag` `\cos` and `\tan` are adjusted to have the same height as `\sin`.

`\sgn` `\arccsc` and other inverse trigonometric functions are defined.

`\Re` `\Im` `\sin` `\cos` `\tan` `\accsc`

1.1 Fractions and units

`\unit` `\inv` The correct spacing for units is provided by the macro `\unit[`⟨*value*⟩`]{`⟨*unit*⟩`}` from the UNITS package [5] which can also be used in text mode. The macro `\inv[`⟨*power*⟩`]{`⟨*text*⟩`}` allows to avoid math mode also for inverse units such as $5\,\mathrm{fb}^{-1}$ typeset via `\unit[5]{\inv{fb}}`.

`\nicefrac` `\flatfrac` `\textfrac` The `\frac{`⟨*number*⟩`}{`⟨*number*⟩`}` macro is accompanied by `\nicefrac{`⟨*number*⟩`}{`⟨*number*⟩`}`, `\textfrac{`⟨*number*⟩`}{`⟨*number*⟩`}`, and `\flatfrac{`⟨*number*⟩`}{`⟨*number*⟩`}` leading to $\frac{1}{2}$, $1/2$, $1/2$, and 1/2. The `\textfrac` macro is mostly intended if a font with oldstyle numerals is used.

*This document corresponds to HEP-MATH v1.2.

†jan.hajer@tecnico.ulisboa.pt

Some macros of the PHYSICS package [6] are reimplemented with a more conventional typesetting in mind. Finer details about mathematical typesetting can be found in [7].

1.2 Differentials and derivatives

\differential
\newderivative
\newpartialderivative
\d
\dv

The three macros `\differential{`⟨*symbol*⟩`}`, `\newderivative {\name}{`⟨*symbol*⟩`}`, and `\newpartialderivative {\name}{`⟨*symbol*⟩`}` allow to define a differential with correct spacing, a derivative using this differential, and if necessary a partial derivative that can handle three dimensional derivatives.

These macros are used for the usual differential and derivative, producing dx via `\d x` and

`\dv[f]x`	`\dv*[f]x^n`	`\dv[f]x*^n`	`\dv*[f]x*^n`
$\frac{df}{dx}$	$d^n f/dx^n$	$\frac{d^n f}{dx^n}$	${}^{d^n f}/_{dx^n}$
`\dv xf`	`\dv*xf`	`\dv x*f`	`\dv*x*f`
$\frac{d}{dx} f$	$d/dx\, f$	$\frac{d}{dx} f$	${}^{d}/_{dx} f$

via `\dv*[`⟨*f*⟩`]{`⟨*x*⟩`}*^{`⟨*n*⟩`}`. Upright differential can be produced via `\renewcommand{\diffsymbol}{\mathrm d}`. The differential takes care of the correct spacing as long as it is placed at the end of the integral $\int f(x)\,dx$. In order to archive correct spacing when it is placed at the beginning of the integral it is advisable to place the whole expression in a `\mathop{\int\d x} f(x)` such that $\int dx\, f(x)$.

\pd
\pdv

Similarly a partial differential and derivative are defined that can be used according to `\pdv*[`⟨*f*⟩`]{`⟨*x*⟩`}*^{`⟨*a*⟩`}[`⟨*y*⟩`]^{`⟨*b*⟩`}[`⟨*z*⟩`]^{`⟨*c*⟩`}`.

`\pdv[f]x`	`\pdv[f]x[y]`	`\pdv[f]x^3`	`\pdv[f]x^2[y]`
$\frac{\partial f}{\partial x}$	$\frac{\partial^2 f}{\partial x \partial y}$	$\frac{\partial^3 f}{\partial x^3}$	$\frac{\partial^3 f}{\partial x^2 \partial y}$
`\pdv[f]x^2[y]^3`	`\pdv[f]x[y]^3`	`\pdv x[y]f`	
$\frac{\partial^5 f}{\partial x^2 \partial y^3}$	$\frac{\partial^4 f}{\partial x \partial y^3}$	$\frac{\partial^2}{\partial x \partial y} f$	

\var

Similarly a functional variation and functional derivative are defined.

\fdv
\cancel
\slashed

The `\cancel{`⟨*characters*⟩`}` macro from the CANCEL package [8] and the `\slashed {`⟨*character*⟩`}` macro from the SLASHED package [9] allow to ~~cancel~~ math and use the Dirac slash notation i.e. $\not{\partial}$, respectively.

1.3 Paired delimiters

\abs
\norm
\eval
\order

`\abs x`	`\norm x`	`\norm[2]x`	`\norm*[2]x`
$\lvert x \rvert$	$\lVert x \rVert$	$\lVert x \rVert_2$	$\left\lVert x \right\rVert_2$

`\order x`	`\eval x__0^\infty`	`\eval* x__0^\infty`
$\mathcal{O}(x)$	$x\vert_0^\infty$	$\left. x \right\vert_0^\infty$

\newpair
\comm
\acomm

The `\newpair{`⟨*name*⟩`}{`⟨*left delim*⟩`} {`⟨*right delim*⟩`}_{`⟨*subscript*⟩`}^{`⟨*superscript*⟩`}` macro is defined and used for the definition of (anti-)commutators and Poisson brackets.

`\pb xy`	`\comm xy`	`\acomm xy`
$\{x, y\}$	$[x, y]$	$\{x, y\}$

They can easily be redefined using e.g. `\newpair\comm\lbrack\rbrack_-`.

`\bra` `\ket` `\braket` `\ketbra` Macros for the bra-ket notation are introduced.

`\bra x`	`\ket x`	`\braket xy`	`\ketbra xy`
$\langle x \,\vert$	$\vert x \rangle$	$\langle x \,\vert\, y \rangle$	$\vert x \rangle\langle y \vert$
`\mel xyz`	`\ev x`	`\ev[\Omega] x`	`\vev x`
$\langle x \,\vert\, y \,\vert\, z \rangle$	$\langle x \rangle$	$\langle \Omega \,\vert\, x \,\vert\, \Omega \rangle$	$\langle 0 \,\vert\, x \,\vert\, 0 \rangle$

`\mel` `\ev` `\vev` `\column` `\row` Macros for row and column vectors are introduced together with a symbol for transpose vectors.

`\column{x,y,z}` $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ `\row{x,y,z}^\trans` $(x,\, y,\, z)^{\intercal}$

2 Environments

`eqnarray` The `eqnarray` environment is depreciated, the `split`, `multline`, `align`, `multlined`, `aligned`, `alignedat`, and `cases` environments of the AMSMATH and MATHTOOLS packages should be used instead.

`equation` Use the `equation` environment for short equations.

```
\begin{equation}
  left = right \ .
\end{equation}
```

$$\boxed{\text{left}} = \boxed{\text{right}}\,. \tag{1}$$

`multline` Use the `multline` environment for longer equations.

```
\begin{multline}
  left = right 1 \\
  + right 2 \ .
\end{multline}
```

$$\boxed{\text{left}} = \boxed{\text{right 1}} \\ \boxed{+\ \text{right 2}}\,. \tag{2}$$

`split` Use the `split` sub environment for equations in which multiple equal signs should be aligned.

```
\begin{equation} \begin{split}
  left &= right 1 \\
  &= right 2 \ .
\end{split} \end{equation}
```

$$\begin{aligned} \boxed{\text{left}} &= \boxed{\text{right 1}} \\ &= \boxed{\text{right 2}}\,. \end{aligned} \tag{3}$$

`align` Use the `align` environment for the vertical alignment and horizontal distribution of multiple equations.

```
\begin{subequations} \begin{align}
  left &= right \ , &
  left &= right \ , \\
  left &= right \ , &
  left &= right \ .
\end{align} \end{subequations}
```

$$\boxed{\text{left}} = \boxed{\text{right}}\,, \quad \boxed{\text{left}} = \boxed{\text{right}}\,, \tag{4a}$$

$$\boxed{\text{left}} = \boxed{\text{right}}\,, \quad \boxed{\text{left}} = \boxed{\text{right}}\,. \tag{4b}$$

`aligned` Use the `aligned` environment within a `equation` environment if the aligned equations should be labeled with a single equation number.

`multlined` Use the `multlined` environment if either `split` or `align` contain very long lines.

```
\begin{equation} \begin{split}
  left &= right 1 \\ &=
  \begin{multlined}[t]
     right 2 \\ + right 3 \ .
  \end{multlined}
\end{split} \end{equation}
```

$$\boxed{\text{left}} = \boxed{\text{right 1}} \\ = \boxed{\text{right 2}} \\ \boxed{\text{+ right 3}} \ . \tag{5}$$

`alignat` Use the `alignat` environment together with the `\mathllap` macro for the alignment of multiple equations with vastly different lengths.

```
\begin{subequations}
\begin{alignat}{2}
  left &= long right && \ , \\
  le. 2 &= ri. 2 \ , &
  \mathllap{le. 3 = ri. 3} & \ .
\end{alignat}
\end{subequations}
```

$$\boxed{\text{left}} = \boxed{\text{long right}} \ , \tag{6a}$$

$$\boxed{\text{le. 2}} = \boxed{\text{ri. 2}} \ , \qquad \boxed{\text{le. 3}} = \boxed{\text{ri. 3}} \ . \tag{6b}$$

As a rule of thumb if you have to use `\notag`, `\nonumber`, or perform manual spacing via `\quad` you are probably using the wrong environment.

A Implementation

<*package>

Load the MATHTOOLS package [1] which loads the AMSMATH package [2]. Allow page breaks within equations if necessary. Adjust the thick and med mu skips slightly.

```
\RequirePackage{mathtools}
\mathtoolsset{centercolon}
\allowdisplaybreaks[1]
\thickmuskip=5mu plus 3mu minus 1mu
\medmuskip=4mu plus 2mu minus 3mu
```

`\mathdef` Define the `\mathdef{`⟨*name*⟩`}[`⟨*arguments*⟩`]{`⟨*macro*⟩`}` macro which (re-)defines macros in math mode only. This macro is implemented using the XPARSE package [10].

```
\RequirePackage{xparse}
\DeclareDocumentCommand{\mathdef}{mO{0}om}{%
  \expandafter\let\csname hep@text\string#1\endcsname=#1
  \expandafter\newcommand\csname hep@math\string#1%
  \IfNoValueTF{#3}{\endcsname[#2]}{\endcsname[#2][#3]}{#4}
  \DeclareRobustCommand#1{%
    \ifmmode
      \expandafter\let\expandafter\next\csname%
      hep@math\string#1\endcsname%
    \else
      \expandafter\let\expandafter\next\csname%
      hep@text\string#1\endcsname%
    \fi
    \next
  }%
```

```
}
```

`\i` Provide an upright imaginary unit in math mode.

```
\newcommand{\imaginaryunit}{\text{i}}
\AtBeginDocument{\mathdef{\i}{\imaginaryunit}}
```

`\overline` Redefine `\overline` to be a text macro using the SOULUTF8 package [4]. Extend it as a math macro with the original definition from the AMSMATH package [2].

```
\RequirePackage{soulutf8}
% \def\overline#1{{\renewcommand{\ULdepth}{-1.9ex}{}\uline{#1}}}
\newcommand\textoverline[1]{{\setul{-1.9ex}{}\ul{#1}}}
\let\overline\textoverline
\DeclareRobustCommand{\over@line}[1]{\@@overline{#1}}
\mathdef{\overline}{\over@line}
\newcommand\hep@widebar[1]{%
  \mkern2.5mu\overline{\mkern-2.5mu#1\mkern-.5mu}\mkern.5mu%
}
\newcommand\widebar[1]{%
  \settowidth{\dimen0}{\ensuremath{#1}}%
  \ifdim\dimen0>.475em\hep@widebar{#1}\else\bar{#1}\fi%
}
```

`\oset` Define a new overset macro `\oset[`⟨*offset*⟩`]{`⟨*over*⟩`}{`⟨*base*⟩`}`

```
\newcommand{\oset}[3][-1pt]{%
  \text{\raisebox{.2ex}{$\mathop{#3}\limits^{%
    \vbox to#1{\kern-2\ex@\hbox{$\scriptscriptstyle#2$}\vss}%
  }$}}%
}
```

`\overleftright` Define a over left right arrow `\overleftright{`⟨*base*⟩`}`.

```
\newcommand{\overleft}[1]{\oset{\leftarrow}{#1}}
\newcommand{\overright}[1]{\oset{\rightarrow}{#1}}
\newcommand{\overleftright}[1]{\oset{\leftrightarrow}{#1}}
```

`eqnarray` Undefine the `eqnarray` environment if not prevented by package option.

```
% \newif\ifhep@eqnarray\hep@eqnarraytrue
% \ifhep@eqnarray\else
%   \let\eqnarray\@undefined
%   \let\endeqnarray\@undefined
% \fi
```

A.1 Operators

`\tr` `\Tr` `\rank` `\erf` `\Res` `\sgn` `\sgn` `\diag` Provide the `\diag`, `\sgn`, and some other operators.

```
\DeclareMathOperator{\tr}{tr}
\DeclareMathOperator{\Tr}{Tr}
```

```
\DeclareMathOperator{\rank}{rank}
\DeclareMathOperator{\erf}{erf}
\DeclareMathOperator{\Res}{Res}
\DeclareMathOperator{\sgn}{sgn}
\DeclareMathOperator{\diag}{diag}
\let\det\relax\DeclareMathOperator{\det}{det}
```

`\Re` `\Im` Redefine the real and imaginary projectors.

```
\let\Re\relax\DeclareMathOperator{\Re}{Re}
\let\Im\relax\DeclareMathOperator{\Im}{Im}
```

`\transpose` `\trans` Define a transpose symbol.

```
\RequirePackage{amssymb}
\newcommand*{\hep@transpose}[2]{\raisebox{\depth}{$\m@th#1\intercal$}}
\newcommand*{\transpose}{{\mathpalette\hep@transpose{}}}
\let\trans\transpose
```

A.1.1 Trigonometric functions

`\cos` `\tan` Adjust the height of of cos and tan to be equal to sin.

```
\let\cos\undefined\DeclareMathOperator{\cos}{cos\vphantom{i}}
\let\tan\undefined\DeclareMathOperator{\tan}{tan\vphantom{i}}
```

`\arccsc` `\arcsec` `\arccot` Define arc operators.

```
\DeclareMathOperator{\arccsc}{arccsc}
\DeclareMathOperator{\arcsec}{arcsec}
\DeclareMathOperator{\arccot}{arccot}
```

`\asin` `\acos` `\atan` `\acsc` `\asec` `\acot` Define shorthand for arc operators.

```
\DeclareMathOperator{\asin}{asin}
\DeclareMathOperator{\acos}{acos}
\DeclareMathOperator{\atan}{atan}
\DeclareMathOperator{\acsc}{acsc}
\DeclareMathOperator{\asec}{asec}
\DeclareMathOperator{\acot}{acot}
```

`\csch` `\sech` Define csch and sech operators.

```
\DeclareMathOperator{\csch}{csch}
\DeclareMathOperator{\sech}{sech}
```

A.2 Units and fractions

`\unit` Load the UNITS package [5] which provides the `\units` and `\nicefrac` macros. Patch the `\unit` macros to behave like `\mathinner` within an equation

```
\RequirePackage{units}
```

```
\let\oldunit\unit
\renewcommand{\unit}[2][]{%
  \ifthenelse{\boolean{mmode}}{%
    \mathinner{\oldunit[#1]{#2}}%
  }{%
     \oldunit[#1]{#2}%
  }%
}
\let\oldunitfrac\unitfrac
\renewcommand{\unitfrac}[3][]{%
  \ifthenelse{\boolean{mmode}}{%
    \mathinner{\oldunitfrac[#1]{#2}{#3}}%
  }{%
    \oldunitfrac[#1]{#2}{#3}%
  }%
}
```

`\inv` Provide a macro for the inverse, useful in combination with the unit macro in text mode.

```
\newcommand{\inv}[2][1]{#2\ensuremath{^{-#1}}}
```

`\textfrac` Provide the `\textfrac` macro useful in combination with a font using lining numerals.

```
\newcommand{\textfrac}[2]{\ensuremath{\nicefrac{\text{#1}}{\text{#2}}}}
```

`\flatfrac` Provide a flat fraction.

```
\DeclarePairedDelimiterX{\hep@flatfrac}[2]{.}{.}{%
  \kern-\nulldelimiterspace#1\delimsize/%
  \hep@left@delim#2\kern-\nulldelimiterspace%
}
\NewDocumentCommand{\flatfrac}{somm}{%
  \mathinner{%
    \IfBooleanTF{#1}{%
      \hep@flatfrac*{#3}{#4}%
    }{%
      \IfNoValueTF{#2}{\hep@left@delim#3/\hep@left@delim#4%
      }{%
       \hep@flatfrac[#2]{#3}{#4}%
      }%
    }%
  }%
}
```

A.2.1 Differentials and derivatives

`\int` Redefine `\int`.

```
\let\hep@int\int
\RenewDocumentCommand{\int}{oe{_^}}{%
  \def\temp{\hep@int_{\IfValueT{#2}{#2}}^{\IfValueT{#3}{#3}}}%
  \IfValueTF{#1}{\mathop{\temp#1}}{\temp}%
```

```
}
```

`\differential` Define a generic differential `\differential`.

```
\newcommand{\differential}[1]{\mathop{}\!#1}
```

`\newderivative` Define a generic derivative.

```
\newcommand\newderivative[2]{
  \NewDocumentCommand{#1}{somse{^}}{%
    \IfBooleanTF{##4}{%
      \IfBooleanTF{##1}{\nicefrac}{\frac}%
    }{%
      \IfBooleanTF{##1}{\flatfrac}{\dfrac}%
    }{%
      \differential#2\IfValueT{##5}{^{##5\!}}\IfValueT{##2}{##2}%
    }{%
      \differential#2{##3}\IfValueT{##5}{^{##5}}%
    }%
  }
}
```

`newpartialderivative` Define a generic partial derivative

```
\newcommand\newpartialderivative[2]{
  \NewDocumentCommand{#1}{somsE{^}{1}oE{^}{1}oE{^}{1}}{%
    \def\hep@one{\IfValueTF{##6}{##7}{0}}
    \def\hep@two{\IfValueTF{##8}{##9}{0}}
    \def\hep@sum{\the\numexpr##5+\hep@one+\hep@two\relax}
    \IfBooleanTF{##4}{%
      \IfBooleanTF{##1}{\nicefrac}{\frac}%
    }{%
      \IfBooleanTF{##1}{\flatfrac}{\dfrac}%
    }{%
      \differential#2\ifnum\hep@sum=1\relax\else{^{\hep@sum\!}}\fi
      \IfValueT{##2}{##2}%
    }{%
      \differential#2{##3}\ifnum##5=1\relax\else{^{##5}}\fi%
      \IfValueT{##6}{#2##6\ifnum##7=1\relax\else{^{##7}}\fi}%
      \IfValueT{##8}{#2##8\ifnum##9=1\relax\else{^{##9}}\fi}%
    }%
  }
}
```

`\diffsymbol` `\diff` `\d` `\derivative` `\dv` Define the differential `\d` and the usual derivative.

```
\providecommand{\diffsymbol}{d}
\newcommand{\diff}{\differential\diffsymbol}
\AtBeginDocument{\mathdef{\d}{\diff}}
\newderivative{\derivative}{\diffsymbol}
\newcommand\dv{\derivative}
```

\partialdifferential Define the partial differential and derivative.
\pd
\partialderivative
\pdv

```
\newcommand\partialdifferential{\differential\partial}
\newcommand\pd{\partialdifferential}
\newpartialderivative{\partialderivative}{\partial}
\newcommand\pdv{\partialderivative}
```

\gaugediffsymbol Define the gauge covariant differential \D.
\gaugediff
\D

```
\providecommand{\gaugediffsymbol}{D}
\newcommand{\gaugediff}{\differential\gaugediffsymbol}
\newcommand{\D}{\gaugediff}
```

\covariantdiff Define the covariant differential \cd.
\cd

```
\newcommand{\covariantdiff}{\differential\nabla}
\newcommand{\cd}{\covariantdiff}
```

\variation Define the functional variation and derivative.
\var
\functionalderivative
\fdv

```
\newcommand\variation{\differential\delta}
\newcommand\var{\variation}
\newpartialderivative{\functionalderivative}{\delta}
\newcommand\fdv{\functionalderivative}
```

\cancel Load the CANCEL [8] and SLASHED [9] packages which provide the \cancel and \slashed macros.
\slashed

```
\RequirePackage{cancel}
\RequirePackage{slashed}
\declareslashed{}{/}{.14}{0}{L}
\declareslashed{}{/}{.06}{0}{\D}
\declareslashed{}{/}{.055}{0}{\pd}
```

A.3 Paired delimiters

\left Load the MLEFTRIGHT package [3] and adjust the spacing around \left and \right.
\right

```
\RequirePackage{mleftright}
\mleftright
```

\noargumentsymbol Allow for macros to have an empty argument using the ETOOLBOX package [11].
\optionalargument

```
\RequirePackage{etoolbox}
\newcommand{\noargumentsymbol}{\:\cdot\:}
\newcommand{\optionalargument}[1]{\ifblank{#1}{\noargumentsymbol}{#1}}
```

\abs Absolute value and norm.
\norm

```
\DeclarePairedDelimiterX\abs[1]\lvert\rvert{\optionalargument{#1}}
\DeclarePairedDelimiterX\hep@norm[1]\lVert\rVert{\optionalargument{#1}}
\DeclarePairedDelimiterXPP\hep@pnorm[2]{}\lVert\rVert{_{#1}}{#2}
\NewDocumentCommand{\norm}{som}{%
```

```
  \IfValueTF{#2}{%
    \IfBooleanTF{#1}{\hep@pnorm*}{\hep@pnorm}{#2}%
  }{%
    \IfBooleanTF{#1}{\hep@norm*}{\hep@norm}%
  }{\optionalargument{#3}}%
}
```

`\floor` `\ceil` Floor and ceiling paired delimiters.

```
\DeclarePairedDelimiter\ceil{\lceil}{\rceil}
\DeclarePairedDelimiter\floor{\lfloor}{\rfloor}
```

`\ordersymbol` `\order` Order symbol and macro.

```
\providecommand{\ordersymbol}{\mathcal{O}}
\DeclarePairedDelimiterXPP\order[1]{\ordersymbol}(){}{#1}
```

`\evaluated` `\eval` Vertical evaluation bar

```
\DeclarePairedDelimiter{\hep@evaluated}{.}{\rvert}
\NewDocumentCommand{\evaluated}{som}{%
  \IfBooleanTF{#1}{%
    \hep@evaluated*{#3}%
  }{%
    \IfNoValueTF{#2}{#3\rvert}{\hep@evaluated[#2]{#3}}%
  }%
}
\newcommand\eval{\evaluated}
```

`\row` `\column` Shortcuts for rows and columns

```
\newcommand*{\rowseperator}{,\,}
\ExplSyntaxOn
\newcommand*{\hep@row}[1]{
 \seq_set_split:Nnn\hep@seq{,}{#1}
 \begin{matrix}\seq_use:Nn\hep@seq{\rowseperator}\end{matrix}
}
\newcommand*{\hep@column}[1]{%
 \seq_set_split:Nnn\hep@seq{,}{#1}%
 \begin{matrix}\seq_use:Nn\hep@seq{\\}\end{matrix}%
}
\ExplSyntaxOff
\DeclarePairedDelimiterX{\row}[1]{(}{)}{\hep@row{#1}}
\NewDocumentCommand{\column}{me{^}e{_}}{%
  \left(\hep@column{#1}\right)%
  \IfValueT{#2}{^{\!\!\!#2}}\IfValueT{#3}{_{\!\!\!#3}}%
}
```

A.3.1 Set and Probability

`\midbar` Define a generic midbar.

```
\newcommand\hep@left@delim{\mathopen{}}
\providecommand{\midbar}[1][]{%
  \nonscript\:#1\vert\allowbreak\nonscript\:\hep@left@delim%
}
```

Check if nfssect-cfr is loaeded and patch the global \set macro into the cfr namespace

```
\RequirePackage{xpatch}
\@ifundefined{exfs@merge@families}{}{%
  \xpatchcmd{\exfs@merge@families}{\set}{\cfr@set}{}{}%
  \xpatchcmd{\exfs@merge@families}{\set}{\cfr@set}{}{}%
  \xpatchcmd{\exfs@merge@families}{\set}{\cfr@set}{}{}%
}%
```

\suchthat
\set

Define a \set macro that allows a midbar via \suchthat.

```
\providecommand\suchthat{\midbar}
\DeclarePairedDelimiterX\set[1]\{\}{%
  \renewcommand\suchthat{\midbar[\delimsize]}#1%
}
```

\probabilitysymbol
\given
\Pr

Redefine the \Pr macro to a macro that takes a \given macro and generates a midbar.

```
\providecommand{\probabilitysymbol}{\operatorname{Pr}}
\providecommand\given{\midbar}
\DeclarePairedDelimiterXPP\hep@Pr[1]{%
  \probabilitysymbol}(){}{%
  \renewcommand\given{\midbar[\delimsize]}#1%
}
\let\Pr\relax
\NewDocumentCommand{\Pr}{so}{%
  \IfValueTF{#2}{%
    \IfBooleanTF{#1}{\hep@Pr*}{\hep@Pr}{#2}%
  }{%
    \probabilitysymbol%
  }%
}
```

A.3.2 Commutators

\newpair

Define the \newpair macro that generates pairs surrounded by brackets.

```
\NewDocumentCommand{\newpair}{mmme{_}e{^}}{%
  \IfNoValueTF{#4}{%
    \IfNoValueTF{#5}{%
      \DeclarePairedDelimiterX{#1}[2]{#2}{#3}%
    }{%
      \DeclarePairedDelimiterXPP{#1}[2]{}{#2}{#3}{^{#5}}%
    }%
  }{%
    \DeclarePairedDelimiterXPP{#1}[2]{}{#2}{#3}{_{#4}}%
  }{%
```

```
    \optionalargument{##1},\optionalargument{##2}%
  }%
}
```

\innerproduct \poissonbracket \pb \commutator \comm \anticommutator \acomm

Poissonbracket, commutator and anti-commutator.

```
\newpair\innerproduct\langle\rangle
\newpair\poissonbracket\lbrace\rbrace
\newpair\commutator\lbrack\rbrack
\newcommand\pb{\poissonbracket}
\newcommand\comm{\commutator}
\newcommand\acomm{\poissonbracket}
```

A.3.3 Bra-ket notation

\braketspace

Define the space within braket notation.

```
% \providecommand\braketspace{\mskip1mu}
\providecommand\braketouterspace{\mskip1mu}
\providecommand\braketinnerspace{\mskip3mu}
\newcommand\hep@midvert{%
  \braketinnerspace\delimsize\vert\braketinnerspace\hep@left@delim%
}
```

\braket

Define the braket macro.

```
\DeclarePairedDelimiterX\braket[2]{\langle}{\rangle}{%
  \braketouterspace#1\hep@midvert#2\braketouterspace%
}
```

\bra

Define the bra macro.

```
\DeclarePairedDelimiterXPP\hep@bra[1]{%
  }{\langle}{\rvert}{\braketinnerspace%
  }{\braketouterspace#1\braketinnerspace%
}
\NewDocumentCommand{\bra}{smt\ket sgt\ketbra sgg}{%
  \IfBooleanTF{#6}{%
    \IfBooleanTF{#1}{\braket*{#2}{#8}}{\braket{#2}{#8}}%
    \IfBooleanTF{#7}{\bra*{#9}}{\bra{#9}}%
  }{
    \IfBooleanTF{#3}{%
      \IfBooleanTF{#1}{\braket*}{%
        \IfBooleanTF{#4}{\braket*}{\braket}}{#2}{#5%
      }%
    }{%
      \IfBooleanTF{#1}{\hep@bra*}{\hep@bra}{#2}%
    }%
  }%
}
```

`\ket` Define the ket macro.

```
\DeclarePairedDelimiterXPP\ket[1]{%
  \braketinnerspace}{\lvert}{\rangle}{%
}{%
  \braketinnerspace\hep@left@delim#1\braketouterspace%
}
```

`\ketbra` Define the ketbra macro.

```
\NewDocumentCommand{\ketbra}{smm}{%
  \IfBooleanTF{#1}{%
    \ket*{#2}\bra*{#3}%
  }{%
    \ket{#2}\bra{#3}%
  }%
}
```

`\matrixelement` `\mel` Define the matrixelement macro.

```
\DeclarePairedDelimiterX\matrixelement[3]{%
  \langle}{\rangle
}{%
  \braketouterspace#1\hep@midvert#2\hep@midvert#3\braketouterspace%
}
\newcommand\matrixel{\matrixelement}
\newcommand\mel{\matrixelement}
```

`\expectationvalue` `\ev` `\vev` Define the expectationvalue and vev macros.

```
\DeclarePairedDelimiterX\hep@expvalue[1]{\langle}{\rangle}{%
  \braketouterspace#1\braketouterspace%
}
\NewDocumentCommand{\expectationvalue}{som}{%
  \IfNoValueTF{#2}{%
    \IfBooleanTF{#1}{\hep@expvalue*}{\hep@expvalue}{#3}%
  }{%
    \IfBooleanTF{#1}{\matrixelement*}{\matrixelement}{#2}{#3}{#2}%
  }%
}
\newcommand\ev{\expectationvalue}
\newcommand\vev[1]{\expectationvalue[0]{#1}}
```

</package>

B Test

<*test>

```
\documentclass{article}

```

```
\usepackage{hep-math}

\begin{document}

\begin{gather}
  \bra{x}\ket{y}
  \braket*{x}{y}\\
  \dv[f]{x}^3
  \pdv[f]{x}[y]^2[z]^3
  \fdv[f]{x}^3[y]\\
  \set{x \suchthat x \in X}
\end{gather}

\end{document}
```

</test>

C Readme

<*readme>

```
# The ‘hep-math‘ package

Extended math macros

## Introduction

The ‘hep-math‘ package provides some additional features beyond the
‘mathtools‘ and ‘amsmath‘ packages.

To use the package place ‘\usepackage{hep-math}‘ in the preamble.

## Author

Jan Hajer

## License

This file may be distributed and/or modified under the conditions of the
‘LaTeX‘ Project Public License, either version 1.3c of this license or
(at your option) any later version. The latest version of this license is
in ‘http://www.latex-project.org/lppl.txt‘ and version 1.3c or later is
part of all distributions of LaTeX version 2005/12/01 or later.
```

</readme>

References

[1] L. Madsen, M. Høgholm, W. Robertson, and J. Wright. ‘The mathtools package: Mathematical tools to use with amsmath’ (2004). CTAN: mathtools.

[2] *LaTeX Team.* 'The amsmath package: AMS mathematical facilities for LaTeX' (1994). CTAN: amsmath. URL: ams.org/tex/amslatex.
[3] H. Oberdiek. 'The mleftright package: Variants of delimiters that act as maths open/close' (2010). CTAN: mleftright.
[4] H. Oberdiek. 'The soulutf8 package: Permit use of UTF-8 characters in soul' (2007). CTAN: soulutf8.
[5] A. Reichert. 'The units and nicefrac packages: Typeset units' (1998). CTAN: units.
[6] S. C. de la Barrera. 'The physics package: Macros supporting the Mathematics of Physics' (2012). CTAN: physics.
[7] E. Gregorio. 'TeX, LaTeX and math' (2020). URL: latex-project.org/publications/2020-egreg-TUB-tb127gregorio-math.pdf.
[8] D. Arseneau. 'The cancel package: Place lines through maths formulae' (2013). CTAN: cancel.
[9] D. Carlisle. 'The slashed package: Put a slash through characters' (1987). CTAN: slashed.
[10] *LaTeX3 Project.* 'The xparse package: A generic document command parser' (1999). CTAN: xparse.
[11] P. Lehman and J. Wright. 'The etoolbox package: e-TeX tools for LaTeX' (2007). CTAN: etoolbox.